IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: Hugh Svendsen et al.   Examiner: Jung W. Kim
Serial No. 10/813,839   Art Unit: 2132
Filed: 03/31/2004
For: **METHOD AND SYSTEM FOR PROVIDING WEB BROWSING THROUGH A
FIREWALL IN A PEER TO PEER NETWORK**

Mail Stop Appeal Brief – Patents
Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Sir:

A **REPLY BRIEF** is filed herewith in response to the Examiner's Answer mailed

October 1, 2008. If any fees are required in association with this Reply Brief, the Director is

hereby authorized to charge them to Deposit Account 50-1732, and consider this a petition

therefor.

REPLY BRIEF

A. Introduction

In response to the Examiner's Answer mailed October 1, 2008 (hereinafter "Examiner's Answer"), the Appellants submit that claims 1-34 are patentable over the references cited in the Final Office Action mailed March 14, 2008 (hereinafter "Final Office Action"). In particular, none of the cited references, either alone or in combination, disclose the feature of translating an HTTP request into a request packet and sending the request packet to a peer server, which is located behind a firewall, as recited in the claims.

B. Rejections

In the Final Office Action, claims 1-4 and 17-20 were rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 6,349,336 B1 to *Sit et al.* (hereinafter "*Sit*"). In the Final Office Action, the Patent Office also rejected claims 5-7, 15, 16, 21-23, and 31-34 under 35 U.S.C. § 103(a) as being unpatentable over *Sit*. Finally, in the Final Office Action, the Patent Office rejected claims 8, 9, 24, and 25 under 35 U.S.C. § 103(a) as being unpatentable over *Sit* in view of U.S. Patent No. 6,917,965 B2 to *Gupta et al.* (hereinafter "*Gupta*"). Claims 10-14 and 26-30 were deemed to be allowable in the Final Office Action if rewritten in independent form; however, the Appellants have not rewritten these claims in independent form.

C. Arguments

Claims 1-4 and 17-20 were rejected under 35 U.S.C. § 102(b) as being anticipated by *Sit*. The Appellants respectfully traverse the rejection.

According to Chapter 2131 of the M.P.E.P., in order to anticipate a claim under 35 U.S.C. § 102, "the reference must teach every element of the claim." The Appellants submit that *Sit* does not teach every element recited in claims 1-4 and 17-20. More specifically, claim 1 recites a method for providing a Web browser running on a computer with access to a peer server located behind a firewall comprising, among other features, in response to a proxy server receiving an HTTP request to access the peer server from the Web browser, "translating the HTTP request into a request packet and sending the request packet to the peer server." Claim 17 includes similar features. The Appellants submit that *Sit* does not disclose or suggest translating an HTTP request into a request packet and sending the request packet to a peer server, which is

located behind a firewall. Instead, *Sit* discloses fooling a firewall in order to pass data to a browser, which is behind the firewall. More specifically, *Sit* discloses wrapping a request sent from the browser 314E to the web server 308I, which is behind the firewall 305, such that, to the firewall 305, the request appears as a response from the browser 314E to a request sent by the web server 308I.[1] As is well known, wrapping includes a header, which precedes encapsulated data and a trailer, which follows the encapsulated data such that the encapsulated data is not viewable to a firewall. Wrapping does not involve translating an HTTP request into a request packet. In fact, *Sit* teaches away from the present invention in that *Sit* discloses fooling a firewall into allowing the transmission of a packet by altering header information such that the packet appears as something it is not, i.e., instead of being a request, the packet appears as a response.

In the Examiner's Answer, the Patent Office responds to this argument by stating that "both the Specification and the Appellant's arguments do not define what elements constitute a 'request packet'; in particular, Appellant does not identify any feature distinguishing a 'request packet' from other types of packets."[2] The Appellants respectfully disagree. Paragraphs [029] and [031] of the Specification disclose that a servlet thread 150 creates a peer request packet 160 from an HTTP request. The Specification also discloses in paragraph [031] that the peer request packet 160 is passed to a peer server 24'. In addition, paragraph [030] of the Specification and Figure 6A disclose that the peer request packet 160 includes a MessageBoxID 162, an HTTP URL 164, multiple HTTP headers 166, and an HTTP Post Data field 168. Paragraph [030] discloses that the HTTP URL 164 is the URL that was requested from a visiting web browser 30 and the HTTP headers 166 are the HTTP headers from the original request from the visiting web browser 30. Thus, the Specification clearly identifies the request packet 160, which is created from an HTTP request, and includes the HTTP URL and the HTTP headers from the original request. The Appellants submit that the request packet 160, which is created from the HTTP request, is not a HTTP request. Instead, the request packet is formed from a HTTP request and is completely different from a HTTP request. Moreover, the language "translating the HTTP request into a request packet and sending the request packet to the peer server" recited in claim 1 is entirely consistent with this definition. Particularly, claim 1 recites that the HTTP request is

---

[1] See *Sit*, col. 7, ll. 50-57
[2] *See* Examiner's Answer, page 9.

3

translated from the HTTP request into a request packet. Moreover, as mentioned above, the request packet is something entirely different from the HTTP request. In contrast, *Sit* discloses that the response includes the HTTP request. Specifically, as mentioned above, *Sit* discloses that a HTTP request is encapsulated such that the HTTP request is not viewable to a firewall.

The Patent Office also indicates that "the term 'translating' is not defined by the Specification; there is no explicit description of a 'translating' process except to identify that a response or request packet is 'created' from an HTTP packet."[3] The Appellants respectfully disagree. Paragraph [020] discloses that a proxy server 36 multiplexes Web traffic in order to enable generic web traffic flow.[4] The proxy server 36 receives incoming HTTP requests from various sources having various formats and translates these requests such that the peer server 24' is able to receive the packet. The proxy server 36 also forwards these requests to the peer servers 24', which are located behind a firewall 36.[5] Moreover, paragraph [021] of the Specification discloses:

> [021]   FIG. 3 is a flow diagram illustrating the process for enabling a Web browser 30 to access the peer server 24' behind a firewall 34. The process begins in step 50 with the peer server 24 registering an outbound socket connection with the proxy server 36. In step 52, all incoming HTTP requests intended for the peer server 24' are redirected to the proxy server 36. In response to receiving a redirected HTTP request in step 54, the proxy server 36 finds the socket connection to the peer server 24', translates the HTTP requests into a **multiplexed protocol comprising a request packet**, and sends the request packet to the peer server 24'. In step 56, the peer node 26 receives the request packet, demultiplexes the request, converts the request packet back into the original HTTP request, and passes the HTTP request to the local Web server 28. In step 58, the peer node 26 receives an HTTP response from Web server 28, converts the HTTP response into a response packet, and sends the response packet to the proxy server 36 over the outbound socket connection. In step 60, the proxy server 36 receives the response packet from the peer server 24', converts the response packet back into the HTTP response, and sends the HTTP response to the requesting web browser 30 (emphasis added).

Furthermore, paragraphs [029] and [030] of the Specification state the following:

> [029]   The process begins in step 200 when the servlet thread 150 in the proxy server 36 receives the HTTP request in the form of a URL from the web browser 30. In step 202, the registration manager 152 checks the server table 70 (see FIG.

---

[3] *See* Examiner's Answer, page 10.
[4] *See* Specification, paragraph [020].
[5] *See* Specification, paragraph [020].

4) to determine if the peer server identified in the requesting URL is registered with the peer server 24', and if so, returns the corresponding peer socket. In step 204, the servlet thread 150 creates a peer request packet 160 from the HTTP request and then passes that packet to the peer manager 154.

[030]   FIG. 6A is a diagram illustrating the contents of a peer request packet 160. In a preferred embodiment, the peer request packet 160 includes a MessageBoxID 162, an HTTP URL 164, multiple HTTP headers 166, and an HTTP Post Data field 168. The MessageBoxID 162 is a unique identifier for correlating peer request packets 162, peer response packets 170, and peer message boxes 156. The HTTP URL 164 is the URL that was requested from the visiting web browser 30. The HTTP Headers 166 is the HTTP headers from the original request from the visiting web browser 30. The HTTP Post Data field 168 contains data for when the request is a POST command, and not a GET command.

According to the specification, a HTTP request is received by a proxy server, where the proxy server translates the HTTP request into another request, which is not a HTTP request. Specifically, paragraphs [029] and [030] clearly state that the peer request packet 160 is created from the HTTP request, where the servlet thread 150 creates the peer request packet 160 from the HTTP request. Moreover, paragraphs [029] and [030] clearly state that the different request packet includes features from the original HTTP request. After translation, the request packet is sent to a peer server. Thus, the proxy server receives requests from various sources, and translates these requests such that a peer server, which is located behind a firewall, may receive the requests.

The Patent Office also states the following:

"nothing in the specification defines what the definitive features of a 'request packet' as recited in the independent claims are; nothing in the Specification defines what relevant features distinguishes a 'request packet' from the packet disclosed in Sit. As such, the limitation 'translating the HTTP request into a request packet' under the broadest reasonable interpretation standard does not appear to be limiting in the sense as argued by the Appellant."[6]

The Appellants respectfully disagree for a number of reasons. First, as noted above, the Specification clearly discloses that the servlet thread 150 creates the peer request packet 160 from an HTTP request, where the peer request packet 160 includes the definitive features of a MessageBoxID 162, an HTTP URL 164, multiple HTTP headers 166, and an HTTP Post Data field 168 formed from the original HTTP request. These features, i.e., forming a different request packet using a URL and HTTP headers from an HTTP request, distinguish over the

_____

[6] See Examiner's Answer, page 10.

5

request packet disclosed in *Sit*. Specifically, as noted above, and, importantly, as acknowledged by the Patent Office, *Sit* discloses that the HTTP request packet is appended with a header and trailer so that the HTTP request packet looks like a response packet.[7] However, no mention is made at all about translating the HTTP request packet itself. Thus, a different request packet is not formed.

Second, the Appellants submit that the Patent Office is impermissibly broadly construing the features recited in claim 1. According to Chapter 2111 of the M.P.E.P., while claims should be given their broadest reasonable interpretation, the interpretation must be "consistent with the specification." The Appellants submit that the Patent Office is not interpreting claim 1 in a manner that is consistent with the Specification. In particular, as previously discussed, paragraph [021] states that the HTTP request is translated into a multiplexed protocol comprising a request packet. Moreover, paragraph [021] discloses that the request packet is sent to a peer server. Accordingly, the Specification explicitly states that an HTTP request is translated into a request packet, not just a packet.

Furthermore, as detailed above, paragraphs [029] and [030] clearly state that the peer request packet 160 is created from the HTTP request. Moreover, paragraphs [029] and [030] clearly state that the different request packet includes features from the original HTTP request. Therefore, the Appellants submit that the Patent Office is interpreting the feature of, in response to a proxy server receiving an HTTP request to access the peer server from the Web browser, "translating the HTTP request into a request packet and sending the request packet to the peer server" in a manner entirely inconsistent with the Specification. For all of the reasons noted above, claims 1 and 17 are patentable over *Sit*. Likewise, claims 2-4, and 18-20, which respectively depend from claims 1 and 17, are patentable for at least the same reasons along with the novel features recited therein.

Claims 5-7, 15, 16, 21-23, and 31-34 were rejected under 35 U.S.C. § 103(a) as being unpatentable over *Sit*. The Appellants respectfully traverse the rejection.

According to Chapter 2143.03 of the M.P.E.P., in order to "establish *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art." The Appellants submit that *Sit* does not disclose all the features recited in claims 5-7, 15, 16, 21-23, and 31-34. As detailed above, *Sit* does not disclose or suggest all the features

---

[7] *See* Examiner's Answer, page 11.

recited in claim 1 or 17, the base claims from which claims 5-7, 15, 16, 21-23, 31, and 32 ultimately depend. Therefore, these claims are patentable over *Sit* for at least the same reasons noted above with respect to claims 1 and 17.

Claim 33 recites a method for providing a web browser comprising, among other features, in response to a proxy server receiving a redirected HTTP request, "translating the HTTP requests into a multiplexed protocol comprising a request packet, and sending the request packet to the peer server." Claim 34 includes similar features. As detailed above, *Sit* does not disclose translating an HTTP request into a request packet and sending the request packet to a peer server.

Claim 33 also recites that, in response to a peer node receiving an HTTP response from the Web server, "translating the HTTP response into a response packet, and sending the response packet to the proxy server." Claim 34 includes similar features. The Appellants have reviewed *Sit* and submit that *Sit* does not disclose that in response to a peer node receiving an HTTP response from the Web server, translating an HTTP packet into a response packet and sending the response packet to a proxy server. For at least this reason and the reasons noted above with reference to claims 33 and 34, these claims are patentable over *Sit*.

Claims 8, 9, 24, and 25 were rejected under 35 U.S.C. § 103(a) as being unpatentable over *Sit* in view of *Gupta*. The Appellants respectfully traverse the rejection. The Appellants submit that neither *Sit* nor *Gupta*, either alone or in combination, discloses or suggests all the features recited in claims 8, 9, 24, and 25. As detailed above, *Sit* does not disclose or suggest all the features recited in claim 1 or 17, the base claims from which claims 8, 9, 24, and 25 ultimately depend. Moreover, *Gupta* does not overcome the previously noted shortcomings of *Sit*. Therefore, claims 8, 9, 24, and 25 are patentable over the cited references for at least the same reasons noted above with respect to claims 1 and 17.

C. Conclusion

As detailed above, the Patent Office has not shown where all the elements of the pending claims are shown in the prior art with sufficient particularity to sustain either an anticipation rejection or an obviousness rejection. In particular, none of the references, either alone or in combination, disclose the feature of translating an HTTP request into a request packet and

7

sending the request packet to a peer server, which is located behind a firewall, as recited in the claims. Accordingly, the pending claims are patentable over the cited references.

Respectfully submitted,

WITHROW & TERRANOVA, P.L.L.C.

By:

Anthony J. Josephson
Registration No. 45,742
100 Regency Forest Drive, Suite 160
Cary, NC 27518
Telephone: (919) 238-2300

Date: December 1, 2008
Attorney Docket: 1104-062